Geospatial Data and Software Reviews

Meg Miller
University of Manitoba

OpenRefine: An Approachable Open Tool to Clean Research Data

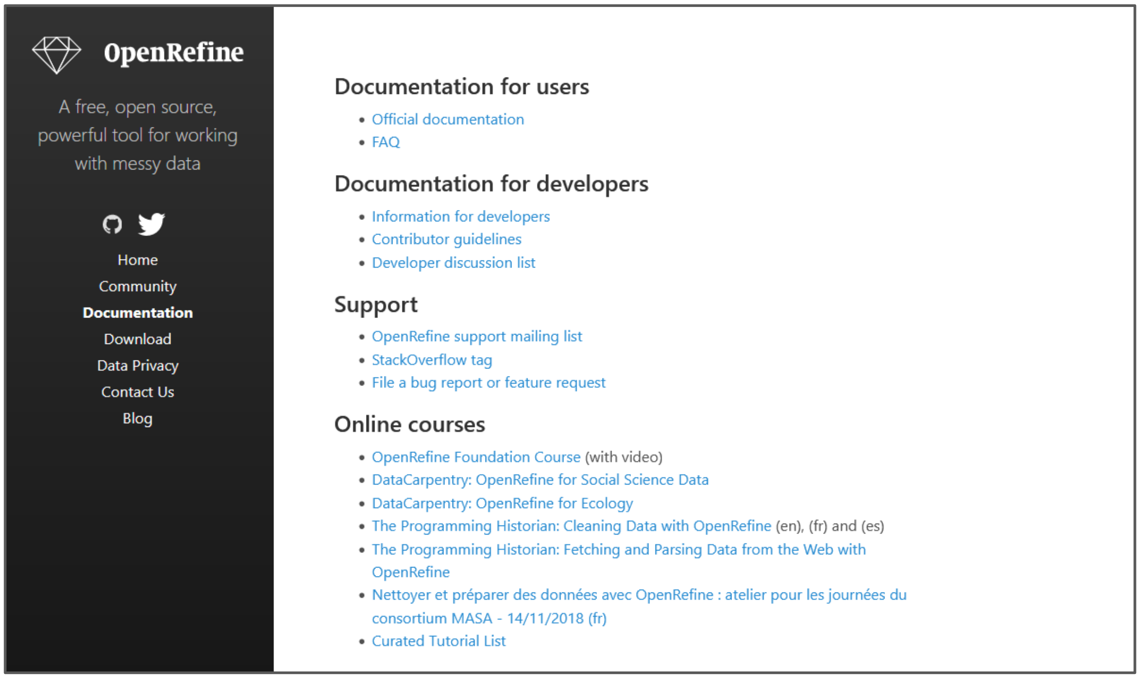
Authors
Meg Miller, GIS & Data Visualization Librarian, University of Manitoba: meg.miller@umanitoba.ca
Natalie Vielfaure, Digital Curation Archivist, University of Manitoba: natalie.vielfaure@umanitoba.ca

Abstract

This review provides an overview of data cleaning tools and discusses why and how OpenRefine has been an effective tool in the delivery of one-shot instructional sessions on data cleaning in an academic library context.

Introduction

The following will examine OpenRefine, an open-source Java based program that runs locally. This application can be used to clean and transform data. The tool is very flexible with a modular flow that lends itself well to on-the-fly modification in a classroom context. Here, the tool is used throughout a session to illustrate different data cleaning techniques and considerations as applied to research data and integrating secondary data sources.



Documentation for users

- Official documentation
- FAQ

Documentation for developers

- Information for developers
- Contributor guidelines
- Developer discussion list

Support

- OpenRefine support mailing list
- StackOverflow tag
- File a bug report or feature request

Online courses

- OpenRefine Foundation Course (with video)
- DataCarpentry: OpenRefine for Social Science Data
- DataCarpentry: OpenRefine for Ecology
- The Programming Historian: Cleaning Data with OpenRefine (en), (fr) and (es)
- The Programming Historian: Fetching and Parsing Data from the Web with OpenRefine
- Nettoyer et préparer des données avec OpenRefine : atelier pour les journées du consortium MASA - 14/11/2018 (fr)
- Curated Tutorial List

Figure 1: OpenRefine documentation landing page

Session Background

Data cleaning sessions were developed in response to user feedback requesting more sessions covering the steps of the data analytics cycle surrounding visualization. Initially Microsoft Excel was the tool of choice for classroom instruction, but as the service grew to encompass more open tools, it was discovered there was an appetite for OpenRefine outside of one-off consultation style support.

At the beginning of the session, survey data that has yet to be cleaned is presented in a wordcloud to quickly highlight problems in the dataset. Another wordcloud will be generated at the end of the session to help users visualize the impact of the data cleaning actions applied in OpenRefine.

Housekeeping (data management vs. data cleaning), faceting, clustering, merging, splitting, appending and more are all covered in relation to researcher data within the session. OpenRefine also allows you to capture a log of all actions taken on a dataset, which is a requirement for some journals and granting agencies to support a move toward open science. This log can also be used to repeat the actions you take on multiple files.

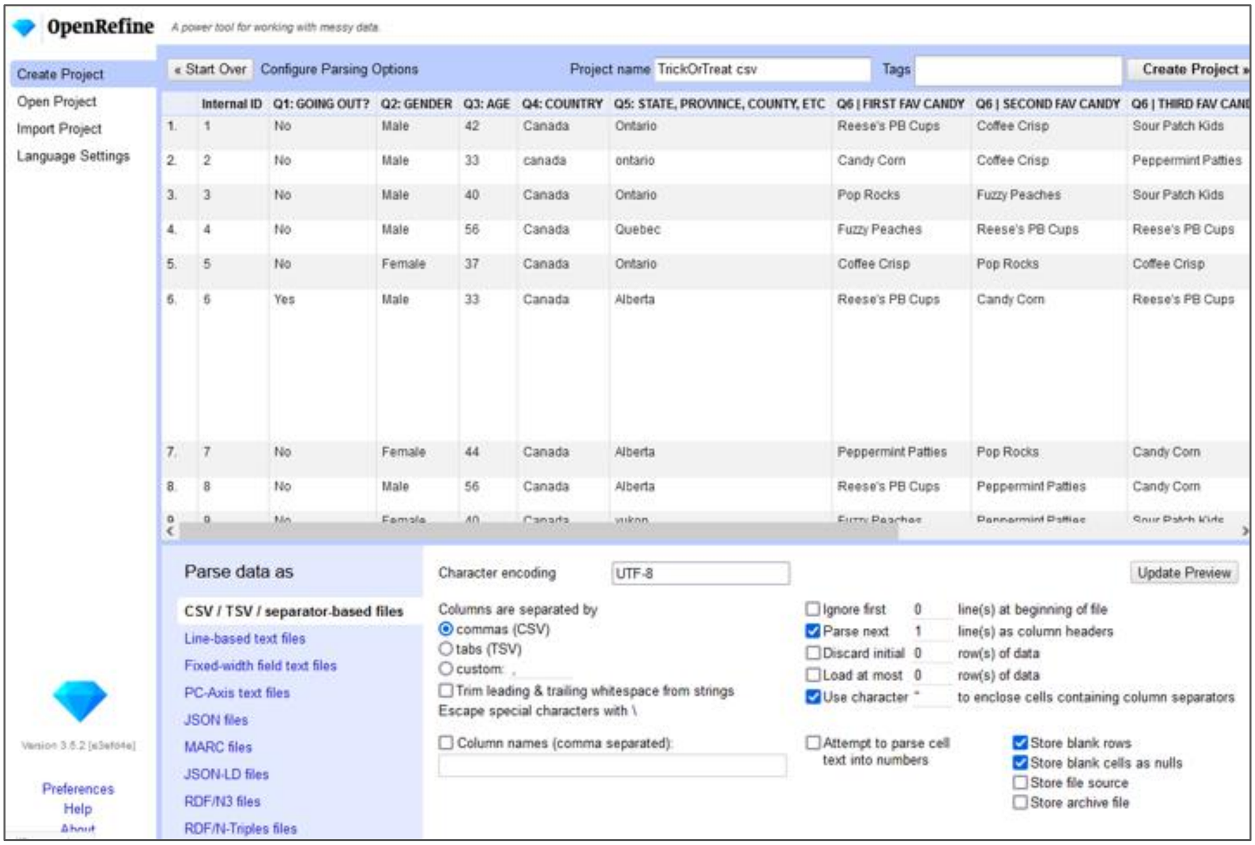
Tool Details

OpenRefine is an open desktop tool for cleaning messy data. It was previously known as Google Refine. In 2012 Google stopped supporting the project and maintenance has been taken over by a dedicated team of volunteers from around the world (https://github.com/OpenRefine/OpenRefine#credits).

OpenRefine looks like a spreadsheet but operates like a database in a web browser. It allows for a low barrier entry point and a data cleaning alternative to Microsoft Excel. The GUI means that novice users can feel confident in applying basic data cleaning strategies and algorithms, while the advanced functions and use of GREL (General Refine Expression Language) allow for more advanced use cases and scenarios.

The application keeps data private on your own computer and works by running a local server that you interact with via your web browser of choice. The GUI looks like a spreadsheet, but operates like a database, allowing for increased discovery capabilities beyond programs like Microsoft Excel. It comes with many pre-loaded transformations, but expressions can be written in General Refine Expression Language (GREL), in Jython (i.e., Python), and in Clojure. There are many extensions, reconciliation services, and client libraries available to users, as well as robust communities of support that can be tapped into via mailing list or StackOverflow.

To start out, users need to download and install OpenRefine on their workstations. The sample dataset is loaded, and series of actions are taken on the dataset. Major categories are listed below, with ones that lend themselves well to simple modification highlighted.



Project name TrickOrTreat csv
Tags

	Internal ID	Q1: GOING OUT?	Q2: GENDER	Q3: AGE	Q4: COUNTRY	Q5: STATE, PROVINCE, COUNTY, ETC	Q6 | FIRST FAV CANDY	Q6 | SECOND FAV CANDY	Q6 | THIRD FAV CAND
1.	1	No	Male	42	Canada	Ontario	Reese's PB Cups	Coffee Crisp	Sour Patch Kids
2.	2	No	Male	33	canada	ontario	Candy Corn	Coffee Crisp	Peppermint Patties
3.	3	No	Male	40	Canada	Ontario	Pop Rocks	Fuzzy Peaches	Sour Patch Kids
4.	4	No	Male	56	Canada	Quebec	Fuzzy Peaches	Reese's PB Cups	Reese's PB Cups
5.	5	No	Female	37	Canada	Ontario	Coffee Crisp	Pop Rocks	Coffee Crisp
6.	6	Yes	Male	33	Canada	Alberta	Reese's PB Cups	Candy Corn	Reese's PB Cups
7.	7	No	Female	44	Canada	Alberta	Peppermint Patties	Pop Rocks	Candy Corn
8.	8	No	Male	56	Canada	Alberta	Reese's PB Cups	Peppermint Patties	Candy Corn
9.	9	No	Female	40	Canada	yukon	Fuzzy Peaches	Peppermint Patties	Sour Patch Kids

Character encoding UTF-8

Columns are separated by
● commas (CSV)
○ tabs (TSV)
○ custom: ,
□ Trim leading & trailing whitespace from strings
Escape special characters with \

□ Column names (comma separated):

□ Ignore first 0 line(s) at beginning of file
☑ Parse next 1 line(s) as column headers
□ Discard initial 0 row(s) of data
□ Load at most 0 row(s) of data
☑ Use character " to enclose cells containing column separators

□ Attempt to parse cell text into numbers

☑ Store blank rows
☑ Store blank cells as nulls
□ Store file source
□ Store archive file

Figure 2: OpenRefine Preview window

Format Support

OpenRefine organizes your data into a project file. When you are transforming the data, it is not the raw data itself that is being manipulated, but the project file. A project file can be created by importing data from your computer, the web, your clipboard, a database, or Google Drive. Many formats are supported, but the most relevant for this audience would be: CSV, TXT, JSON, XML, XLS, MARC and RDF.

Housekeeping

If you supply two or more files for one project, the files' rows will be loaded in the order that you specify, and OpenRefine will create a column at the beginning of the dataset populated with the source URL or file name to help you identify where each row came from. If the files have columns with identical names, the data will load in those columns; if not, the successive files will append all their new columns to the end of the dataset.

If your dataset is more than one million rows you will need to increase your RAM and the server memory allocated. Additionally, the software you are importing from or exporting to may have a different set of limitations to consider.

This is a useful place to discuss data conversions, residency, field types, and the importance of planning one's project.

Dataset structure – Columns

Actions that would be taken on a column include deleting, renaming, adding, hiding, sorting and moving columns. More advanced transformations like splitting and joining columns will be covered in a later section.

This is a good starting point for users to familiarize themselves with navigating the program interface.

Data contents – Cells

Common cell transformations would include trimming whitespace, changing type case, field types and null values. In OpenRefine, changing values to title case is as simple as clicking the dropdown arrow in the header row for the column in question (1), selecting *Edit cells* (2), then *Common transforms* (3), and finally *To titlecase* (4). Other transformations that can be done using GREL are found under the *Transform* option.

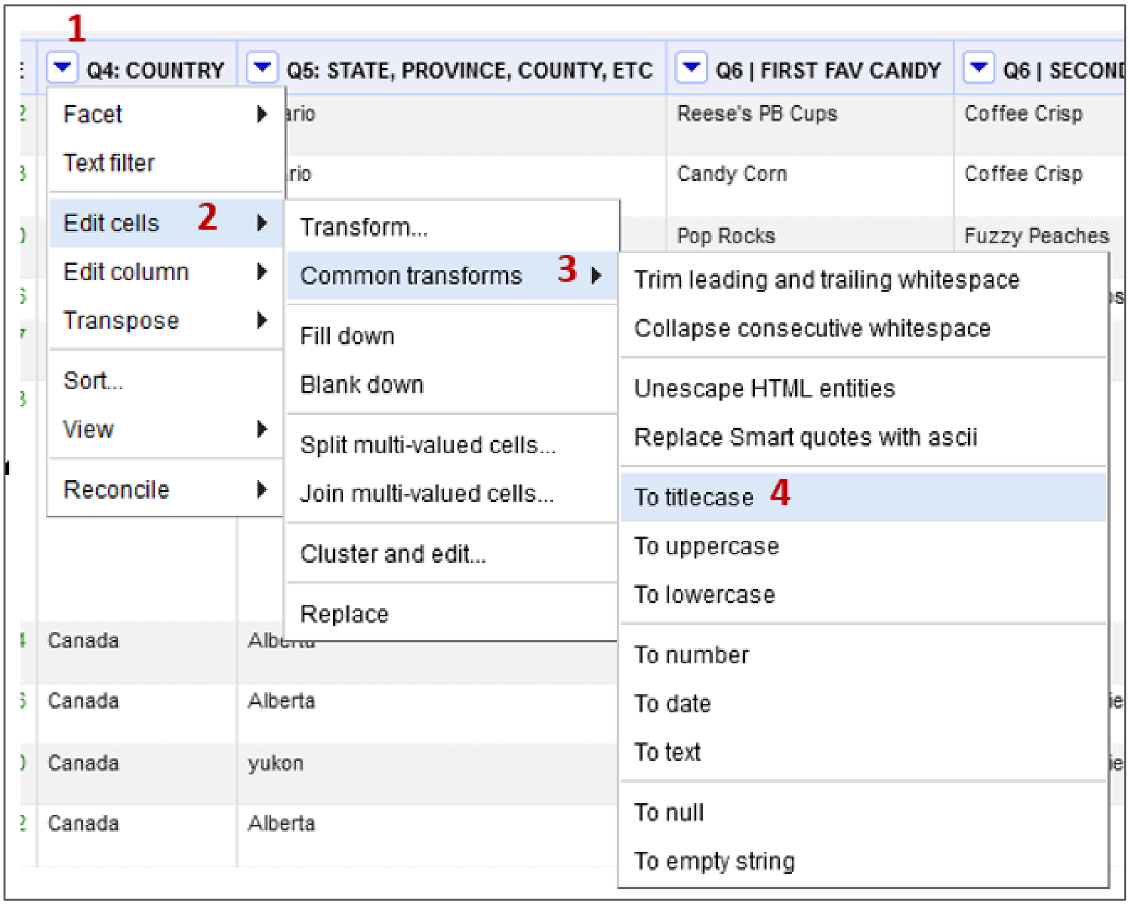
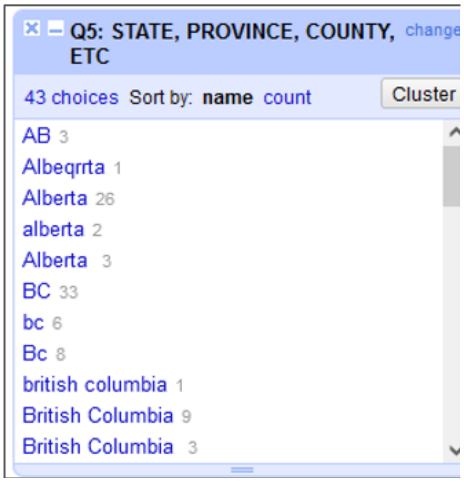


Figure 3: Workflow to change values to title case

Facets

Faceting provides the user with a snapshot of the entries in a particular column and allows them to filter down to a particular record. I use it most to quickly highlight problems with the data and look for outliers. The facet provides a list of cells in a given column to better assess the big picture for that column and further allows you to filter to some subset of rows for which the cells in that column satisfy some constraint.

Edits to the data can be manually made from the Facet pane, where values can be sorted alphabetically (name) or by frequency (count). In an instructional session, this section transitions well into more advanced transformations such as clustering.



Q5: STATE, PROVINCE, COUNTY, ETC

43 choices Sort by: name count

AB	3
Albeqrrta	1
Alberta	26
alberta	2
Alberta	3
BC	33
bc	6
Bc	8
british columbia	1
British Columbia	9
British Columbia	3

Figure 4: Facet view of sample data column

Popular transformations

This section discusses the most popular transformations for which researchers request support. In the session, discussion focusses on not blindly trusting the technology and being an active/considered participant in the data cleaning process.

Cluster

Clustering uses AI and fuzzy matching to best guess which pieces of text refer to the same thing and highlight potential inconsistencies that could be resolved. Opportunity is given to play with the *Method* and *Keying Function* to change the type of algorithm used to fuzzy match. The program does a great job of matching the correct values, but as with anything automated, users are reminded to go through and look at their data for any discrepancies after applying a change. Workflows are discussed at this point: Depending on the data set, what makes the most sense? Running a different clustering algorithm or merging the remaining clusters by hand.

Join

There are many different reasons a user might want to merge or join multiple columns together. Discussion here usually centers around matching field types, and column sizes so records do not get truncated.

Split

Splitting is the most common transformation users request support with, either for parsing text or separating coordinate pairs for plotting data on a map. Working through an example and demonstrating how to use the facets to display records that did not split properly is useful, as you aren’t just demonstrating an action, but how to troubleshoot. This example further reminds users of the importance of data cleaning, as consistently structured data will make the splitting process easier.

Extensions

Many plugins have been developed for OpenRefine. The ones that are most relevant to someone in a role supporting data visualization in an academic setting are:

- GeoRefine: Adds GIS functionality to GREL library
- OSM Extractor: Imports OpenStreetMap data using Overpass API
- GeoJSON Export: Exports data as GeoJSON using coordinate pairs or WKT

Note that some operations will involve connecting to an external service. The interface does a decent job at documenting this, but user discretion is advised. A rule of thumb if you are using a service for reconciliation or connecting to an external URL (database, Google Drive, Geocoder...), is read the documentation about how your data is being stored.

In a classroom context, no time is spent actively exploring these plugins. Instead, discussion is centered on seeking help in discussion forums, not reinventing the wheel, and knowing when to step back.

Many examples from each of these actions and more are available on the project site with the code heavily commented to explain what is going on. A link to the documentation can be found at https://docs.openrefine.org/ .

Session workflow (1-1.5hrs)

How does this play out in an instructional session?

Before the session:

1. Students were instructed to download the most stable long-term release of the software and have it installed on their machines before the session.
2. Learning materials (dataset, slides, walkthrough, and FAQs) are uploaded to the GitHub workshop space.

The session:

Topic	***Action***
Session introduction	*Participants are reminded to download and install materials if they have forgotten.*
Overview of Data Cleaning and Software	*Through a slide deck, participants are provided with a definition of data cleaning, common steps that may be applied to a data set to improve its overall quality, and overall benefits of data cleaning for the researcher and secondary users of the data. Participants are then given an introduction to the OpenRefine, as well as Voyant, which will be used to generate a wordcloud.*
Introduce Example Scenario	*Participants are cleaning up survey data to be imported into a GIS software.*
OpenRefine Exercise	
Housekeeping & Importing	*Participants examine the contents of the data folder and preview data using OpenRefine.*

Cell Transformations • To titlecase • To number	*Participants convert:* • *Country column to titlecase* • *Age field from string to number*
Faceting & Clustering	*Participants apply a facet to the Province column to explore the state of the data. Manual editing and outliers are discussed.* *Participants explore the options available in the clustering dialogue and the limitations.*
Column Transformations • Merge • Split • Rename • Delete	*Participants:* • *Merge 3 columns together using a separator* • *Split a coordinate pair column into two columns and trim whitespace* • *Rename the split columns to have meaningful names* • *Delete an empty column (and discuss why hiding can be a better option)*
Advanced Techniques	*Participants discuss their advanced needs, facilitator discusses troubleshooting and where to go for support (documentation, support forums etc).*
Exporting	*Participants export cleaned data as a CSV file and visualize in a wordcloud software such as Voyant or WordArt.com and discuss further steps.*
Wrap-up	*Participants are provided with information on library supports for more specific data cleaning questions or issues. The remaining time is used to take questions from the audience.*

Discussion

In talking to stakeholders, it was found there was a lot of interest on and off campus for sessions on data cleaning. Outside of the typical audience, the Digital Curation Archivist and I attracted a diverse group when the session was offered as a GradSteps session. We have also been invited to run the session as a brown-bag to the Manitoba GIS User Group- a provincial community of practice of GIS users from government, academia, non-profits and private industry.

For the future, intermediate level sessions without canned data will be offered as well as targeted offerings for the archival studies program in the history department.

For archives, OpenRefine offers many opportunities to more effectively manage metadata, accession databases, and archival descriptions. Formats supported by OpenRefine, such as CSV, TXT, and, XML, lend themselves well to this type of work. Cleaning up data exported from a legacy database ahead of ingesting it to new or upgraded databases, flagging inconsistencies in descriptive metadata, or correcting errors in text generated through OCR for digitized records can be resource-intensive tasks. OpenRefine can help reduce headaches in this process by automating some of this work to ensure better long-term access and preservation of holdings.

Overall, OpenRefine's ability to perform some of the heavy lifting for the user makes it worthwhile tool to add to an archivist's, librarian's or researcher's data resource toolkit.